文章编号:2095-476X(2014)05-0060-04

基于Hadoop的云存储系统的设计与研究

刘姝

(中原工学院 计算机学院, 河南 郑州 450007)

摘要:针对海量数据的存储和处理,设计了一个基于Hadoop的云存储系统.该系统在分布式文件系统和MapReduce编程模型2个核心技术的基础上建立基于Hadoop的云存储模型,优化了存储方式,提高了集群中网络带宽和磁盘的利用率,同时MapReduce编程框架的设计使系统拥有更强的计算能力.该系统可通过Linux集群技术搭建Hadoop平台,进行测试和分析.应用实践表明,该系统具有低成本、高效率、易扩展和安全可靠等特点,能稳定高效地满足海量数据的处理要求.

关键词:云存储;Hadoop;分布式并行计算;HDFS;MapReduce

中图分类号:TP311 **文献标志码**:A DOI:10.3969/j.issn.2095-476X.2014.05.014

Design and research of cloud storage system based on Hadoop

LIU Shu

(*College of Computer Science, Zhongyuan University of Technology, Zhengzhou* 450007, *China*)

Abstract: Aiming at the problem of mass data storage and processing, a cloud storage system based on Hadoop was designed. The system established a cloud storage model on the basis of studying the two core technologies of Hadoop including Hadoop distributed file system and programming model MapReduce. The system optimized the computer storage mode, and increased the efficiency of the network bandwidth and disk in the cluster. At the same time, the MapReduce programming framework design made the system have higher performance of computing ability. Through testing and analysis of Hadoop platform using Linux cluster technology, the results showed that the system had the characteristics of low cost, high efficiency, easy extension, safty and reliability, and could meet the requirement of the mass data processing stably and efficiently.

Key words: cloud storage; Hadoop; distributed parallel computing; HDFS; MapReduce

0 引言

随着网络技术的快速发展,信息资源日益膨胀,这些信息来自电信、金融、教育、医疗等多个行业,不仅包含文字数据,而且涵盖了图片、音频、视频等各种多媒体数据.传统的存储方式硬件要求高,编写并行程序困难,已经无法满足以指数级速度增长的数据存储与处理需求.

2006年Google公司率先提出了云计算概念[1],这是一种创新的计算商业模式,它融合并发展了分布式计算、网格计算、并行处理、网络存储以及虚拟化和负载均衡等多种计算机技术和网络技术.云计算中的"云"是一个包含硬件和软件的虚拟化资源池,这个资源池在网络的平台上为用户提供各种服务,而这些服务建立在各种标准和协议之上,用户可以随时随地申请所需的服务,在资源池中各种资

收稿日期:2014-04-25

基金项目:河南省科技攻关项目(132102310284)

作者简介:刘姝(1975—),女,江苏省兴化市人,中原工学院讲师,硕士,主要研究方向为数字资源管理、网络安全.

源协同工作,使存储和计算海量数据成为可能.

Hadoop 是 Apache 组织下的一个开源项目,是参考了 Google 公司云计算 3 大核心技术 GFS,Mapreduce 和 Bigtable 的设计思想开发的一个分布式云计算平台[2-8]. 该平台是一个用 JAVA 实现的进行数据处理和数据分析的分布式并行编程框架,部署到由大量低廉的硬件设备组成的计算机集群上,主要专注于海量数据的存储和计算,不仅可以处理结构化数据、日志文件以及点击流数据,还可以管理类似 Facebook 的非结构化文本数据[3]. 本文对 Hadoop 的主要组成部分分布式文件系统 HDFS 和编程模型 MapReduce 进行深入研究,拟提出一个基于 Hadoop 的云计算存储模型,以稳定高效地满足海量数据的处理要求.

1　系统分析与设计

1.1　基于 Hadoop 的云存储模型

根据对 Hadoop 的分布式文件系统与计算模型的研究,提出了基于 Hadoop 的云存储模型,如图 1 所示.

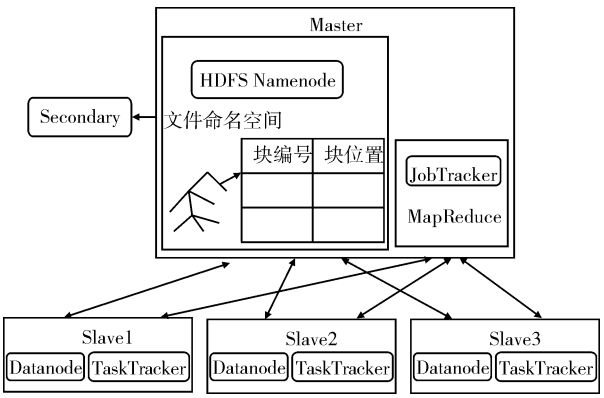

图 1　Hadoop 云存储模型

模型包括 1 个主节点 Master 和若干个子节点 Slave. Master 的工作通常由性能较高的服务器来完成,主要是分布式文件系统的命名空间的维护,并通过 JobTracker 管理其他机器上的 TaskTracker,为其分配相应的 Map 和 Reduce 任务. 每一个子节点不仅是存储数据节点,也是计算节点,因此将 Datanode 和 TaskTracker 的功能集中在 1 台 Slave 完成. 为了避免当 Master 发生故障造成系统服务中断甚至完全瘫痪,设置了 1 台 Secondary Master 作为备份主节点.

在应用该模型的过程中,将系统需要的原始数据存储到分布式文件系统中,在 Master 的 Namenode 中建立文件系统的索引目录以管理分别存储于不同 Datanode 中的数据块. 客户向系统提交任务请求,提交的任务实际上是一组自定义的 API,即用户自定义的 Map 和 Reduce 函数. JobTracker 负责向所有的 Slave 分发 Map 函数和 Reduce 函数,依照 MapReduce 的执行过程产生最终结果.

1.2　系统架构设计

整个云存储系统建立在 Hadoop 存储模型基础之上,由 1 台 Master 和多台 Slave 组成,将原始的数据资源分布式存储于 HDFS 中,HDFS 支持批量处理,同时在 Hadoop 的计算模型中支持移动计算. Map 与 Reduce 函数并行,执行完成查询与计算任务. 该架构可提高系统吞吐量并降低网络传输压力.

系统平台整体架构如图 2 所示. 客户端通过 Web 浏览器和一些应用软件向系统提出各种请求,系统响应客户端的请求,调用相关的应用服务,主要包括访问控制、数据检索、资源管理、安全备份、Web 服务等. 在涉及到庞大的数据量和复杂计算的情况下,使用 1 台服务器无法满足高效率的要求,因此和 Hadoop 平台相结合完成用户的请求,并将处理的结果返回给客户端. 该架构具有可扩展性,可以将关系数据库整合进来,通过去异构化操作为用户透明地提供各种存储和应用服务.

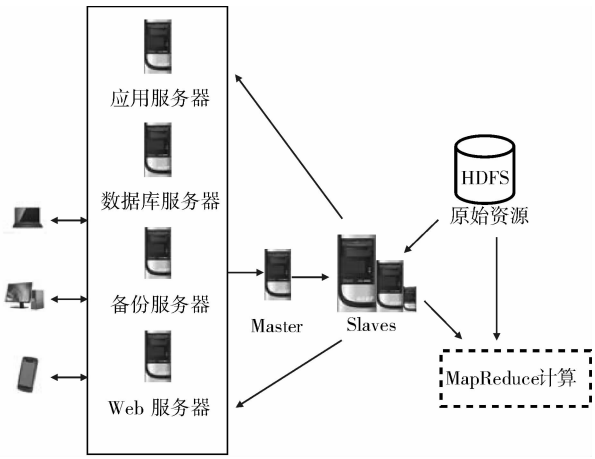

图 2　Hadoop 云存储系统平台整体架构

1.3　系统功能设计

从云存储系统需要实现之功能的角度考虑,整个体系自下而上包括 3 层结构,分别是基础设施管理层、数据存储管理层和应用接口层,如图 3 所示.

基础设施管理层位于整个功能框架的最底层,是云存储的基础. 在该层上将各种不同类型的存储设备进行连接,形成一个统一的存储设备管理体系,并且利用 HDFS 实现海量数据的存储虚拟化,同

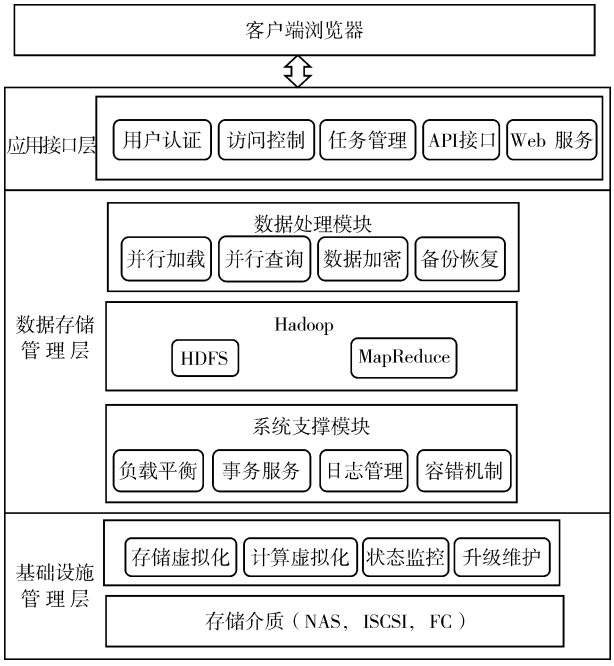

图 3 系统功能结构图

时结合服务器集群虚拟计算和网络进行整合,形成一个动态化的资源池,从而实现了所有资源的集中管理以及集群的状态监控和维护升级,大大提高了资源共享率及扩展性.

数据存储管理层是整个系统的核心部分,实现了底层存储和上层访问的无缝联接. 在该层中采用了分布式文件系统、分布式并行计算和 Linux 集群等技术,可以并行处理海量数据,同时还提供了系统支撑模块,保证了系统的正常运行. 系统支撑模块通过负载平衡、日志管理和容错机制等为实现分布式管理提供了支撑服务. 负载平衡用来存储集群中各节点的负载均衡,并进行容错管理. 日志管理用来记录软件运行过程中每一点的状态、发生的重要事件以及系统的运行轨道. 此外,还需对集群的远程配置、系统状态监测和维护提供服务. 在系统支撑模块的保证下,依据 Hadoop 平台提供的相关组件和技术,可以实现对海量数据的并行加载和并行查询等功能,同时为了提高系统的安全性,加强了对存储和传送数据过程中的数据加密以及对数据存储的备份管理和备份恢复.

应用接口层根据需要的实际业务类型以及提供的不同服务,开发出相关的应用程序,并与底层集群相联接,集群接收到客户端提交的数据并进行处理,将计算结果通过服务器返回给客户端. 用户可以在客户端通过简便友好的操作界面存储和处理海量数据;也可以基于算法库中的公共 API 编写程序,扩展应用系统的功能. 为了增强系统的扩展性和透明性,在访问数据库时需要对数据源进行屏蔽.

2 系统搭建与性能测试

2.1 Hadoop 集群搭建

1)环境配置. 系统采用 6 台 PC 机搭建起底层 Hadoop 集群. 每台机器的操作系统为 CentOS 5.5, JAVA 环境为 jdk1.6,Hadoop 版本为 0.20.2.

选择 1 台机器作为集群的主节点 Master,担当起 Namenode 及 JobTracker 的任务,其他几台机器作为从节点 Slave,完成 Datanode 与 TaskTracker 的功能. 每台 PC 机的 IP 地址与名称如下:

192.168.0.1 Node1
192.168.0.2 Node2
192.168.0.3 Node3
192.168.0.4 Node4
192.168.0.5 Node5
192.168.0.6 Node6

要确保系统正确地解析各主机名与 IP 地址,需要对各机器进行相应的配置. 其中 Node1 代表 Master,在该机器的/etc/hosts 文件中添加集群内部所有机器对应的 IP 地址和主机名. Node2—6 代表 Slave,只需要在/etc/hosts 文件中添加本机和 Master 的 IP 地址及主机名.

2)建立 SSH 免密码登录. 在 Hadoop 集群中,不同机器之间的通信是通过 SSH 来实现的,对 SSH 进行配置的目的是保证网络的畅通,在机器之间通信执行指令时不再需要输入密码,从而建立 SSH 受信证书. 在主节点利用 ssh-keygen 创建 Master 的密钥对.

```
$ ssh-keygen-t rsa-p
$ cp /home/. ssh/id_rsa. pub > > home/. ssh/authorized_keys
```

将 Master 上创建并保存的密钥对通过 scp 命令拷贝到 Slave 相同的目录中,即可实现集群中机器之间的 SSH 免密码登录.

3)搭建 Hadoop 平台. ① 对 conf/hadoop_env. sh 文件中的 2 个环境变量 JAVA_HOME 和 HADOOP_HOME 进行设置,其中 JAVA_HOME 设置为安装 JAVA 的根目录.

② 编辑 Hadoop 的配置文件 core-site. xml,hdfs-site. xml 与 mapred-site. xml. 其中前 2 个文件与 HDFS 的配置相关,主要是设置 Namenode 的 IP 端

口以及数据备份数量;第 3 个文件与 MapReduce 相关,用来配置 JobTracker 的 IP 及端口.

③ 编辑 conf 目录下的 masters 和 slaves 文件. 将 masters 文件的内容设置为主节点的 IP 地址 192.168.0.1 或机器名 Node1,将 slaves 文件的内容添加为所有从节点的 IP 地址或机器名,如 192.168.0.2—6,每个 IP 地址占一行.

④ 把在主节点配置好的 hadoop-0.20.2 目录使用 scp 命令从 master 发送到各个 slaves,命令形式如下:

$ scp-r home/hadoop-0.20.2 node2@192.168.0.2:/home/

⑤ 将分布式文件系统 HDFS 格式化,并启动所有守护进程.

$ bin/hadoop namenode-format

$ bin/start-all.sh

2.2　平台性能测试和分析

在搭建 Hadoop 集群的基础上,采用本文提出的系统功能架构,可以实现相应的任务管理、数据管理以及集群管理等功能模块,相关的操作诸如上传数据、下载数据等,可以调用 Hadoop 的 API 函数来完成. 通过 Hadoop 的内置 Web 接口可以查看整个系统的工作进展情况以及集群的当前工作状态. 利用主节点 192.168.0.1 中不同端口的设置,可以访问 Namenode 的状态、Map 函数和 Reduce 函数的运行情况,以及浏览 HDFS 中的文件内容和日志信息.

对系统进行性能测试,将测试数据划分为不同的数量级别,集群与单机性能对比情况见表 1.

表 1　集群与单机性能测试对比

数据量 /GB	耗费时间 T_n/s		T_1/T_6
	$n=1$	$n=6$	
1	62.592	225.153	0.278
3	213.569	195.631	1.092
5	313.762	235.578	1.332
10	685.135	266.768	2.568
20	1 395.375	288.752	4.832

由表 1 可知: 1)随着处理数据量的增大,单机运算所耗费的时间成倍地增长,性能下降明显;而 Hadoop 集群所需时间在一定范围内仅出现小幅变化,并且远远低于单机所用时间,显示出了集群在处理大规模数据时的优越性. 2)当数据量较少时,集群所耗费的时间比单机耗费时间多,处理效率低于单机,这是由于当系统启动并初始化时需要消耗相应的资源,集群内部机器彼此交互中间文件时也会耗费一定的时间. 当数据规模明显增加后,集群这些消耗在整个系统业务量中所占的比例明显下降,而单机会随着数据规模的增加出现效率瓶颈.

3　结语

本文设计了一个基于 Hadoop 的云存储框架,并搭建起了 Hadoop 云存储平台,通过分布式文件系统 HDFS 和 MapReduce 计算模型等相关技术对海量数据进行处理. 该系统具有低成本、高容错性和安全性、高扩展性和高效率等特点,解决了数据扩充出现的瓶颈问题,拥有更高性能的计算能力,可以广泛地应用于学校、机关和企事业单位等不同领域. 但作为一种新兴的处理海量数据的模式,云存储本身还有一些不完善和急需解决的问题,系统的功能也需要进一步完善,如 Hadoop 中如何解决存储和计算的紧耦合问题、如何实现存储和计算的有效分离问题、异构数据库的整合问题等.

参考文献:

[1] Boutaba R, Cheng L, Zhang Q. On cloud computational models and the heterogeneity challenge[J]. Journal of Internet Services and Applications, 2012, 3(1): 77.

[2] 周品. Hadoop 云计算实战[M]. 北京:清华大学出版社, 2012.

[3] 翟岩龙,罗壮,杨凯,等. 基于 Hadoop 的高性能海量数据处理平台研究[J]. 计算机科学, 2013, 40(3): 100.

[4] 马林山,赵庆峰,肖新国. 基于 Hadoop 的云移动信息服务模型研究[J]. 情报科学, 2013, 31(4): 28.

[5] Bass L, Kazman R, Ozkaya I. Open Source Systems: Grounding Research[M]. Berlin: Springer, 2011: 50 - 61.

[6] [美]Lam C. Hadoop 实战[M]. 韩冀中,译. 北京:人民邮电出版社, 2011.

[7] 亢丽芸,王效岳,白如江. MapReduce 原理及其主要实现平台分析[J]. 现代图书情报技术, 2012(2): 60.

[8] 李寒,唐兴兴. 基于参数优化的 Hadoop 云计算平台[J]. 计算机系统应用, 2013, 22(3): 21.